orgmode examples

draw, code evaluation and present in orgmode with LaTeX beamer

kimim

Document ID: Revision: Updated on: September 26, 2021

Outline

orgmode examples

Introduction

Purpose

To evaluate many features of orgmode, such as

- drawing with code
- evaluating results of code snippets
- exporting orgmode file to pdf slides

Introduction

How

Following tools are used in this file:

- MSYS2 provides many tools and libraries
- GraalVM JDK supports Java, JS, R and more
- GNU Emacs with kimim-emacs configuration
- Org Mode, including org-babel, org-export
- TexLive with beamertheme-kimim style
- PlantUML, Graphviz, LaTeX tikz package
- Inkscape to convert svg to pdf, during orgmode-pdf exporting

orgmode examples

Preparation

Emacs settings

You may need to use kimim-emacs configuration:

```
# backup existing emacs config
cd ~ && mv .emacs .emacs-backup && mv .emacs.d .emacs.d-backup
# clone this config
git clone https://github.com/kimim/kimim-emacs
# copy default .emacs to ~
cp kimim-emacs/.emacs ~
```


Preparation

Emacs and Orgmode version

Firstly, let's check GNU Emacs[1] and Orgmode[2] version:

```
(concat  (emacs-version)
         "\nOrgmode " (org-version))
```

```
GNU Emacs 28.0.50 (build 6, x86_64-w64-mingw32)
 of 2021-08-31
Orgmode 9.4.4
```

[1] https://www.gnu.org/software/emacs

[2] https://orgmode.org

Preparation

TexLive and Beamer Theme

Install TexLive[3] to `<texlive-path>` and clone beamertheme-kimim[4], and update TEX cache:

```
git clone https://github.com/kimim/beamertheme-kimim \
    <texlive-path>/texmf-local/tex/latex/beamertheme-kimim
mktexlsr
```

[3] http://tug.org/texlive

[4] https://github.com/kimim/beamertheme-kimim

Preparation

Inkscape version

Install Inkscape[5] to convert SVG image to PDF.

This is inkscape version on my Windows 10:

```
inkscape --version
```

```
Inkscape 1.0.2-2 (e86c870879, 2021-01-15)
```

[5] https://inkscape.org

orgmode examples

Drawing in code

PlantUML settings in Emacs

Download plantuml.jar[6], and set jar-path

```
(require 'url-handlers)
(require 'ob-plantuml)
(url-copy-file "https://nchc.dl.sourceforge.net/project/plantuml/plantuml.jar"
               "./plantuml.jar" t)
(setq org-plantuml-jar-path "./plantuml.jar")
```

[6] https://plantuml.com

Drawing in code

PlantUML version

Here is the version info on my machine, including JVM, dot and graphviz:

```
(shell-command-to-string
 (concat
  "java -jar " org-plantuml-jar-path " -version"))
```

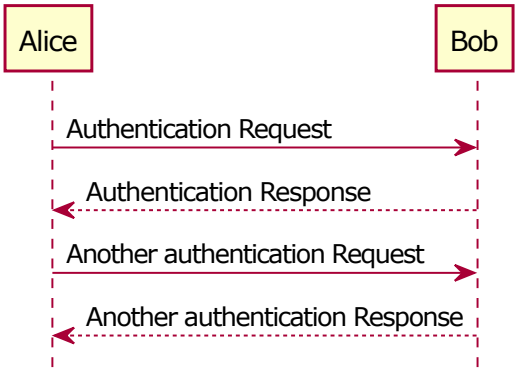
```
PlantUML version 1.2021.8 (Sat Jun 26 16:20:59 CST 2021)
(GPL source distribution)
Java Runtime: OpenJDK Runtime Environment
JVM: OpenJDK 64-Bit Server VM
Default Encoding: Cp1252
Language: en
Country: US

PLANTUML_LIMIT_SIZE: 4096

Dot version: dot - graphviz version 2.44.1 (20200629.0846)
Installation seems OK. File generation OK
```


Drawing in code

Sequence Diagram

Let's draw a simple sequence diagram with this plantuml code:

```
@startuml
hide footbox
hide unlinked
Alice -> Bob: Authentication Request
Bob --> Alice: Authentication Response
Alice -> Bob: Another authentication Request
Alice <-- Bob: Another authentication Response
@enduml
```

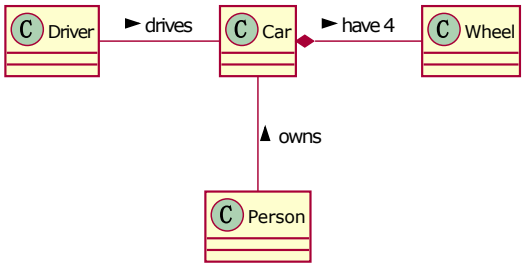


Drawing in code

Class Diagram

A simple class diagram

```
@startuml
class Car
Driver - Car : drives >
Car *- Wheel : have 4 >
Car -- Person : < owns
@enduml
```



Drawing in code

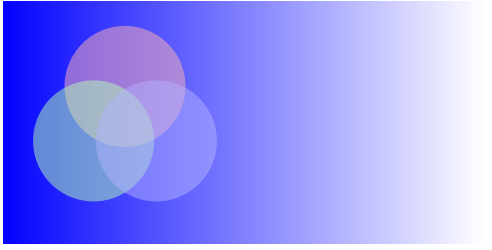
tikz logo

```
\begin{tikzpicture}
  \filldraw[blue] (0,0) rectangle (-4,-2);
  \filldraw[blue,rotate=-30] (0,0) rectangle (1,-0.2);
  \filldraw[blue] (-4,0) circle (0.2);
  \filldraw[blue] (-4,-2) rectangle (-3,-3);
  \filldraw[blue] (0,-2) rectangle (-1,-3);
  \filldraw[blue] (-4,0) rectangle (-5.5,-1.5);
\end{tikzpicture}
```

Drawing in code

tikz background

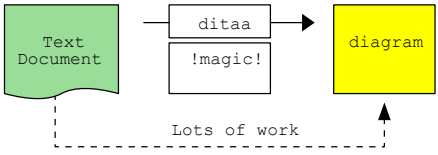
```
\begin{tikzpicture}
  \fill[left color=blue,right color=white,shading angle=90,line width=0] (0,0) rectangle (4,-2);
  \begin{scope}[shift={(1,-1)}]
    \fill[red!30!white,semitransparent]   ( 90:0.3) circle (0.5);
    \fill[green!30!white,semitransparent] (210:0.3) circle (0.5);
    \fill[blue!30!white,semitransparent]  (330:0.3) circle (0.5);
  \end{scope}
\end{tikzpicture}
```



Drawing in code

ditaa

```
+--------+   +-------+    +-------+
|cGRE    | --+ ditaa +--> |cFF0   |
|  Text  |   +-------+    |diagram|
|Document|   |!magic!|    |       |
|     {d}|   |       |    |       |
+---+----+   +-------+    +-------+
    :                         ^
    |       Lots of work      |
    +-------------------------+
```

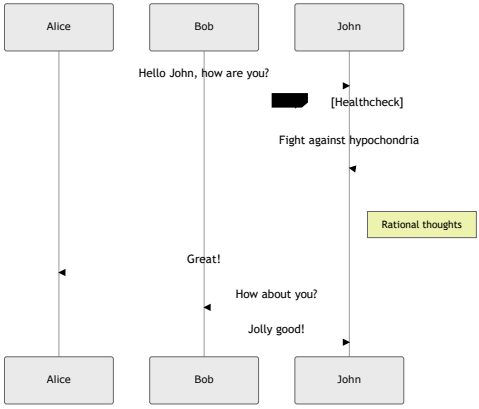


Drawing in code

mermaid

Installation and Configuration, see
kimim-emacs#mermaid

```
sequenceDiagram
    participant Alice
    participant Bob
    Alice->>John: Hello John, how are you?
    loop Healthcheck
        John->>John: Fight against hypochondria
    end
    Note right of John: Rational thoughts
    John-->>Alice: Great!
    John->>Bob: How about you?
    Bob-->>John: Jolly good!
```



orgmode examples

Org-babel Evaluating Programming Languages

emacs lisp

```
(emacs-version)
```

```
GNU Emacs 28.0.50 (build 6, x86_64-w64-mingw32)
 of 2021-08-31
```

```
(decoded-time-year (decode-time (current-time)))
```

```
2021
```


Org-babel Evaluating Programming Languages

shell

```
sh --version
```

```
GNU bash, version 5.1.8(1)-release (x86_64-pc-msys)
Copyright (C) 2020 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>

This is free software; you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.
```


Org-babel Evaluating Programming Languages

C

```c
printf("%s is %d years old\n", "C programming language", year-1972);
```

```
C programming language is 49 years old
```


Org-babel Evaluating Programming Languages

C++

```
cout << "C++ is " << year-1979 << " years old" << endl;
```

```
C++ is 42 years old
```


Org-babel Evaluating Programming Languages

Clojure

```
(println "Clojure is" (- year 2005) "years old")
```

```
Clojure is 16 years old
```


Org-babel Evaluating Programming Languages

ClojureScript

TODO

Org-babel Evaluating Programming Languages

Java

TODO: can pass variable to java

```
public class Main{
    public static void main(String[] args){
        System.out.println("Java is " + (2021-1995) + " years old");
        return;
    }
}
```

```
Java is 26 years old
```


Org-babel Evaluating Programming Languages

Python

Check Python version in shell:

```
python --version
```

```
Python 3.9.6
```

Evaluate Python code:

```
print("Python is " + str(year - 1991) + " years old")
```

```
Python is 30 years old
```


Org-babel Evaluating Programming Languages

Rust

```
(package-install 'ob-rust)
```

TODO: cannot pass variable to rust

```
fn main() {
    println!("Rust is {} years old", 2021 - 2016);
}
```


Org-babel Evaluating Programming Languages

Go

TODO

```
package main
import ("fmt")

func main(){
   fmt.Println("emacs")
}
```


Org-babel Evaluating Programming Languages

R

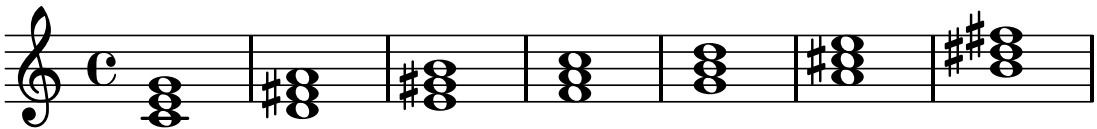
TODO

orgmode examples

Org-babel for music and others

LilyPond

```
\relative c' {
  \chordmode {c1}
  \chordmode {d1}
  \chordmode {e1}
  \chordmode {f1}
  \chordmode {g1}
  \chordmode {a1}
  \chordmode {b1}
}
```



Org-babel for music and others

Math Equations

orgmode examples

Presenting with Org-beamer

Beamer

In this section, I will try some beamer settings in orgmode.

Presenting with Org-beamer

latexmk version

```
(princ (concat (format "LaTeXmk version: %s\n"
                       (eshell-command-result "latexmk --version") "\n")
               (format "XeTeX version: %s\n"
                       (eshell-command-result "xelatex --version") "\n")))
```

```
LaTeXmk version: Latexmk, John Collins, 29 May 2021. Version 4.74b
XeTeX version: XeTeX 3.141592653-2.6-0.999993 (TeX Live 2021/W32TeX)
kpathsea version 6.3.3
Copyright 2021 SIL International, Jonathan Kew and Khaled Hosny.
There is NO warranty.  Redistribution of this software is
covered by the terms of both the XeTeX copyright and
the Lesser GNU General Public License.
For more information about these matters, see the file
named COPYING and the XeTeX source.
Primary author of XeTeX: Jonathan Kew.
Compiled with ICU version 68.2; using 68.2
Compiled with zlib version 1.2.11; using 1.2.11
Compiled with FreeType2 version 2.10.4; using 2.10.4
Compiled with Graphite2 version 1.3.14; using 1.3.14
Compiled with HarfBuzz version 2.7.4; using 2.7.4
Compiled with libpng version 1.6.37; using 1.6.37
Compiled with pplib version v2.05 less toxic i hope
Compiled with fontconfig version 2.13.93; using 2.13.93
```


Presenting with Org-beamer

simple slide

This is a simple slide, with some formatted texts:

- important underline *slashed* `code` `verbatim` ~~deleted~~ alert
  - important underline *slashed* `code` `verbatim` ~~deleted~~ alert
  - important underline *slashed* `code` `verbatim` ~~deleted~~ alert
    - important underline *slashed* `code` `verbatim` ~~deleted~~ alert
    - important underline *slashed* `code` `verbatim` ~~deleted~~ alert
    - important underline *slashed* `code` `verbatim` ~~deleted~~ alert

Enumerations:

1. important underline *slashed* `code` `verbatim` ~~deleted~~ alert

1.1 important underline *slashed* `code` `verbatim` ~~deleted~~ alert

1.2 important underline *slashed* `code` `verbatim` ~~deleted~~ alert

1.2.1 important underline *slashed* `code` `verbatim` ~~deleted~~ alert

1.2.2 important underline *slashed* `code` `verbatim` ~~deleted~~ alert

1.2.3 important underline *slashed* `code` `verbatim` ~~deleted~~ alert

Presenting with Org-beamer

simple slide with definition

It is not recommended to have second level definition bullet…

Beamer
: LaTeX package to generate slides

Orgmode
: Powerful plain text format

org-babel
: Let Orgmode understand and evaluate programming languages

ox-latex
: Exporter to export orgmode to latex and further to PDF

Presenting with Org-beamer

simple slide with wallpaper

- This slide has a nice wallpaper.
- It is the westlake in the morning.

Presenting with Org-beamer

some todo list

- daily task [33%]
  - ☒ fetch the milk in the morning
  - ☐ check the mailbox
  - ☐ clean the garden
- learning task [50%]
  - ☒ read the book
  - ☒ write the reading notes
  - ☐ make a presentation
  - ☐ present to students

Presenting with Org-beamer

table

enrollment to the class	name	date
x	Kimi	2021-09-18
	Ivy	2021-09-28
x	Anna	2021-09-20

Presenting with Org-beamer

4 dimension

up-left

- 1

up-right

- 2

down-left

- 3

down-right

- 4

Presenting with Org-beamer

quotation

Quote:

> *If winter comes, can Spring be far behind?*

Quotation:

> *History repeats itself, and that's one of the things that's wrong with history.*

orgmode examples

Conclusion

Key Takeaways

- Emacs is a long lasting, and wonderful text editor
- Orgmode is an awesome plain text format
- LaTeX is great typesetting tool
- Beamer is a LaTeX package for preparing presentation
- Thus, using these tools within Emacs is cool!

Appendix

References I